Array sorting exercise

1. Declare an **int** array of 1000 elements. Populate it with random numbers in a range from 0 to 500. Display the first 100 elements of the array in original order. Sort it using any sorting algorithms. Display the first 100 elements of the sorted array (you display only the first 100 elements because to print 1000 will take too much space. We can see whether the array is sorted or not by looking at the first hundred elements). Print the time in milliseconds it took to sort the array. Populate the array with random numbers again. Sort it using a different sorting algorithm. Print the time it took to sort the array.

Use
System.currentTimeMillis() - This method returns the difference, measured in **milliseconds**, between the current **time** and **midnight, January 1, 1970**

or
System.nanoTime(); - This method returns the difference, measured in **nanoseconds**, between the current **time** and **midnight, January 1, 1970**

For example:

```
long start=0L, end =0L;          //we pick number of type long not integer
int microseconds=0;              // because the number will be too big to fit in int


start = System.nanoTime();
  // do array sorting
  //here
end = System.nanoTime();

microseconds = (end - start) / 1000;
```